

In step 706, the file transfer procedure 126 receives the input_args parameter.

In step 708, the file transfer procedure 126 uses the pointers in the sqlvars of the input_args parameter to retrieve the file name(s) of the file(s) associated with the procedure 112. The use of pointers to reference memory areas, and to retrieve data from such memory areas, is well known.

In step 710, the file transfer procedure 126 uses the pointers in the sqlvars of the input_args parameter to retrieve the string representation(s) of the file(s) associated with the procedure 112.

In step 712, the file transfer procedure 126 uses the pointers in the sqlvars of the input_args parameter to retrieve the target path.

In step 714, the file transfer procedure 126 writes the string representation(s) (retrieved in step 710) to file(s) (preferably one file per string representation). The procedure for writing a string to a file is analogous to the procedure for converting a file to a string (described above), and will be apparent to persons skilled in the relevant art(s). The files are written under the file names retrieved in step 708. The file format is implementation dependent.

While various embodiments of the present invention have been described above, it should be understood that they have been presented by way of example only, and not limitation. Thus, the breadth and scope of the present invention should not be limited by any of the above-described exemplary embodiments, but should be defined only in accordance with the following claims and their equivalents.

Having thus described our invention, what we claim as new and desire to secure by Letters Patent is:

SUB A1

1. A method of transferring a file from a client platform to a server platform, the server platform coupled to the client platform, a "DB2" relational database management system (RDBMS) executing on the server platform, the method comprising the steps of:

- (1) at the client platform, converting the file to a string;
- (2) at the client platform, causing a pointer in an input_args parameter to point to said string, said input_args parameter being of a "sqllda" data type, said pointer being part of a "sqlvar" parameter defined by the sqllda data type, the sqllda data type being suitable for passing scalar values to procedures; and
- (3) at the client platform, invoking a "sqlproc" function provided by a "client application enablement" (CAE) module, the CAE module being resident in the client platform and representing a client component of the DB2 RDBMS, a parameter list of said sqlproc function comprising said input_args parameter and a procedure_to_invoke parameter that identifies an user-specified procedure located on the server platform, said sqlproc function when executed causing said input_args parameter to be passed to said user-specified procedure at the server platform, and also causing said user-specified procedure to be invoked at the server platform.

2. A method of transferring a file from a client platform to a server platform, the client platform coupled to the server platform, a "DB2" relational database management system (RDBMS) executing on the server platform, the method comprising the steps of:

- (1) at the client platform, converting the file to a string;
- (2) at the client platform, setting a procedure_to_invoke parameter equal to information identifying a file transfer procedure located at the server platform;

- (3) at the client platform, causing a pointer in a first "sqlvar" parameter to point to said string, and causing a pointer in a second sqlvar parameter to point to a file name of the file, the first and second sqlvar parameters being part of an input_args parameter, the input_args parameter having a "sqlda" data type, the sqlda data type being suitable for passing scalar values to procedures; and
- (4) at the client platform, invoking a "sqleproc" function provided by a "client application enablement" (CAE) module, the CAE module being resident in the client platform and representing a client component of the DB2 RDBMS, a parameter list of said sqleproc function comprising said input_args parameter and said procedure_to_invoke parameter, said sqleproc function when executed causing said input_args parameter to be passed to said file transfer procedure at the server platform, and also causing said file transfer procedure to be invoked at the server platform.
3. The method of claim 2, further comprising the steps of:
- (5) at the server platform, receiving the input_args parameter;
- (6) at the server platform, using said pointer in said first sqlvar parameter of the input_args parameter to access and retrieve said string;
- (7) at the server platform, using said pointer in said second sqlvar parameter of the input_args parameter to access and retrieve said file name; and
- (8) at the server platform, writing said string to a new file in the server platform using said file name.
4. The method of claim 3, further comprising the step of:
- (9) at the client platform, causing a pointer in a third sqlvar parameter to point to a target path string.
5. The method of claim 4, further comprising the step of:
- (10) at the server platform, using said pointer in said third sqlvar parameter of the input_args parameter to access and retrieve said target path string;
- wherein step (8) comprises the step of:
- at the server platform, writing said string to a new file using said file name at a location in said server platform identified by said target path string.
6. A method of transferring a file from a client platform to a server platform, the client platform coupled to the server platform, a "DB2" relational database management system (RDBMS) executing on the server platform, the method comprising the steps of:
- (1) at the server platform, receiving an input_args parameter being of a "sqlda" data type, said input_args parameter comprising a "sqlvar" parameter defined by said sqlda data type, said sqlvar parameter comprising a pointer pointing to a string that was transferred from the client platform to the server platform, said string being a string representation of a file located on the client platform;
- (2) at the server platform, using said pointer in said sqlvar parameter of the input_args parameter to access and retrieve said string;
- (3) at the server platform, writing said string to a new file.
7. The method of claim 6, wherein said input_args parameter comprises a second sqlvar parameter having a pointer that points to a file name of said file, and a third sqlvar parameter having a pointer that points to a target path string, the method further comprising the steps of:
- (4) at the server platform, using said pointer in said second sqlvar parameter of the input_args parameter to access and retrieve said file name; and

(5) at the server platform, using said pointer in said third sqlvar parameter of the input_args parameter to access and retrieve said target path string;

wherein step (3) comprises the step of:

at the server platform, writing said string to a new file 5
using said file name at a location in said server platform identified by said target path string.

8. A method for enabling an user to distribute a procedure from a client platform to a plurality of server platforms, and for enabling the user to make and register the procedure at 10
the server platforms, a "DB2" relational database management system (RDBMS) executing on each server platform, comprising the steps of:

(1) at the client platform, displaying a list of servers;
(2) at the client platform, enabling the user to select any 15
of said servers;

(3) at the client platform, commanding a "client application enablement" (CAE) module to distribute a file associated with the procedure to the selected servers, the CAE module being resident in the client platform and representing a client component of the DB2 RDBMS; and 20

(4) at the client platform, commanding the CAE module to cause the selected servers to make and register the procedure; 25

wherein step (3) comprises the steps of:

(a) converting the file to a string;
(b) setting a procedure_to_invoke parameter equal to information identifying a file transfer procedure located at each selected server; 30

(c) causing a pointer in a first "sqlvar" parameter to point to said string, and causing a pointer in a second sqlvar parameter to point to a file name of the file, the first and second sqlvar parameters being part of an input_args parameter, the input_args parameter having a "sqllda" data type, the sqllda data type being suitable for passing scalar values to procedures; and 35

(d) invoking for each selected server a "sqleproc" function provided by the CAE module, a parameter list of said sqleproc function comprising said input_args parameter and said procedure_to_invoke parameter, said sqleproc function when executed causing said input_args parameter to be passed to said file transfer procedure at each selected server, and also causing said file transfer procedure to be invoked at each selected server. 45

9. The method of claim 8, further comprising the steps of:

(5) at each server, receiving the input_args parameter;
(6) at each server, using said pointer in said first sqlvar parameter of the input_args parameter to access and retrieve said string; 50

(7) at each server, using said pointer in said second sqlvar parameter of the input_args parameter to access and retrieve said file name; and 55

(8) at each server, writing said string to a new file in said each server using said file name.

10. The method of claim 8, wherein step (4) comprises the step of:

setting the procedure_to_invoke parameter equal to information identifying a make function located at each selected server; and 60

invoking for each selected server the "sqleproc" function, said sqleproc function when executed causing each selected server to invoke the make function, the make function when executed operating to make the procedure. 65

55449-9433333

11. The method of claim 8, wherein step (4) comprises the step of:

setting the procedure_to_invoke parameter equal to information identifying a registration function located at each selected server; and

invoking for each selected server the "sqlproc" function, said sqlproc function when executed causing each selected server to invoke the registration function, the registration function when executed operating to register the procedure with the DB2 RDBMS at said each selected server.

12. A system of transferring a file from a client platform to a server platform, the client platform coupled to the server platform, a "DB2" relational database management system (RDBMS) executing on the server platform, the system comprising:

means, at the client platform, for converting the file to a string;

means, at the client platform, for setting a procedure_to_invoke parameter equal to information identifying a file transfer procedure located at the server platform;

means, at the client platform, for causing a pointer in a first "sqlvar" parameter to point to said string, and causing a pointer in a second sqlvar parameter to point to a file name of the file, the first and second sqlvar parameters being part of an input_args parameter, the input_args parameter having a "sqllda" data type, the sqllda data type being suitable for passing scalar values to procedures; and

means, at the client platform, for invoking a "sqlproc" function provided by a "client application enablement" (CAE) module, the CAE module being resident in the client platform and representing a client component of the DB2 RDBMS, a parameter list of said sqlproc function comprising said input_args parameter and said procedure_to_invoke parameter, said sqlproc function when executed causing said input_args parameter to be passed to said file transfer procedure at the server platform, and also causing said file transfer procedure to be invoked at the server platform.

13. The system of claim 12, further comprising:

means, at the server platform, for receiving the input_args parameter;

means, at the server platform, for using said pointer in said first sqlvar parameter of the input_args parameter to access and retrieve said string;

means, at the server platform, for using said pointer in said second sqlvar parameter of the input_args parameter to access and retrieve said file name; and

writing means, at the server platform, for writing said string to a new file in the server platform using said file name.

14. The system of claim 13, further comprising:

means, at the client platform, for causing a pointer in a third sqlvar parameter to point to a target path string.

15. The system of claim 14, further comprising:

means, at the server platform, for using said pointer in said third sqlvar parameter of the input_args parameter to access and retrieve said target path string;

wherein said writing means includes means for writing said string to a new file using said file name at a location in said server platform identified by said target path string.

16. A server platform coupled to a client platform, a "DB2" relational database management system (RDBMS) executing on the server platform, the server platform comprising:

means for receiving an input_args parameter being of a "sqlda" data type, said input_args parameter comprising a "sqlvar" parameter defined by said sqlda data type, said sqlvar parameter comprising a pointer pointing to a string that was transferred from the client platform to the server platform, said string being a string representation of a file located on the client platform;

means for using said pointer in said sqlvar parameter of the input_args parameter to access and retrieve said string; and

writing means for writing said string to a new file in the server platform.

17. The server platform of claim 16, wherein said input_args parameter comprises a second sqlvar parameter having a pointer that points to a file name of said file, and a third sqlvar parameter having a pointer that points to a target path string, the server platform further comprising:

means for using said pointer in said second sqlvar parameter of the input_args parameter to access and retrieve said file name; and

means for using said pointer in said third sqlvar parameter of the input_args parameter to access and retrieve said target path string;

wherein said writing means comprises means for writing said new file using said file name at a location in said server platform identified by said target path string.

18. A database system for enabling an user to distribute a procedure from a client platform to a plurality of server platforms, and for enabling the user to make and register the procedure at the server platforms, a "DB2" relational database management system (RDBMS) executing on each server platform, the database system comprising:

means, at the client platform, for displaying a list of servers;

means, at the client platform, for enabling the user to select any of said servers;

CAE commanding means, at the client platform, for commanding a "client application enablement" (CAE) module to distribute a file associated with the procedure to the selected servers, the CAE module being resident in the client platform and representing a client component of the DB2 RDBMS; and

make and register means, at the client platform, for commanding the CAE module to cause the selected servers to make and register the procedure;

wherein said CAE commanding means comprises:

means for converting the file to a string;

means for setting a procedure_to_invoke parameter equal to information identifying a file transfer procedure located at each selected server;

means for causing a pointer in a first "sqlvar" parameter to point to said string, and causing a pointer in a second sqlvar parameter to point to a file name of the file, the first and second sqlvar parameters being part of an input_args parameter, the input_args parameter having a "sqlda" data type, the sqlda data type being suitable for passing scalar values to procedures; and

means for invoking for each selected server a "sqleproc" function provided by the CAE module, a parameter list of said sqleproc function comprising said input_args parameter and said procedure_to_invoke parameter, said sqleproc function when executed causing said input_args parameter to be

6949-929366

passed to said file transfer procedure at each selected server, and also causing said file transfer procedure to be invoked at each selected server.

19. The database system of claim 18, further comprising:

5 means, at each server, for receiving the input_args parameter;

means, at each server, for using said pointer in said first sqlvar parameter of the input_args parameter to access and retrieve said string;

10 means, at each server, for using said pointer in said second sqlvar parameter of the input_args parameter to access and retrieve said file name; and

15 means, at each server, for writing said string to a new file in the server using said file name.

20. The system of claim 18, wherein said make and register means comprises:

20 means for setting the procedure_to_invoke parameter equal to information identifying a make function located at each selected server; and

25 means for invoking for each selected server the "sqlproc" function, said sqlproc function when executed causing each selected server to invoke the make function, the make function when executed operating to make the procedure.

21. The system of claim 18, wherein said make and register means comprises:

30 means for setting the procedure_to_invoke parameter equal to information identifying a registration function located at each selected server; and

35 means for invoking for each selected server the "sqlproc" function, said sqlproc function when executed causing each selected server to invoke the registration function, the registration function when executed operating to register the procedure with the DB2 RDBMS at said each selected server.

22. A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform method steps for transferring a file from a client platform to a server platform, the server platform coupled to the client platform, a "DB2" relational database management system (RDBMS) executing on the server platform, said method steps comprising:

45 (1) at the client platform, converting the file to a string;

(2) at the client platform, causing a pointer in an input_args parameter to point to said string, said input_args parameter being of a "sqllda" data type, said pointer being part of a "sqlvar" parameter defined by the sqllda data type, the sqllda data type being suitable for passing scalar values to procedures; and

50 (3) at the client platform, invoking a "sqlproc" function provided by a "client application enablement" (CAE) module, the CAE module being resident in the client platform and representing a client component of the DB2 RDBMS, a parameter list of said sqlproc function comprising said input_args parameter and a procedure_to_invoke parameter that identifies an user-specified procedure located on the server platform, said sqlproc function when executed causing said input_args parameter to be passed to said user-specified procedure at the server platform, and also causing said user-specified procedure to be invoked at the server platform.

65 23. A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform method steps for transferring a file

552349-0233333

- (1) at the client platform, converting the file to a string;
- (2) at the client platform, setting a procedure_to_invoke parameter equal to information identifying a file transfer procedure located at the server platform;
- (3) at the client platform, causing a pointer in a first "sqlvar" parameter to point to said string, and causing a pointer in a second sqlvar parameter to point to a file name of the file, the first and second sqlvar parameters being part of an input_args parameter, the input_args parameter having a "sqlda" data type, the sqlda data type being suitable for passing scalar values to procedures; and

24. The program storage device of claim 23, said method steps further comprising the steps of: 30

(6) at the server platform, using said pointer in said first sqlvar parameter of the input_args parameter to access and retrieve said string; 35

(8) at the server platform, writing said string to a new file in the server platform using said file name.

(9) at the client platform, causing a pointer in a third sqlvar parameter to point to a target path string.

(10) at the server platform, using said pointer in said third sqlvar parameter of the input_args parameter to access and retrieve said target path string;

27. A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform method steps for transferring a file from a client platform to a server platform, the client platform coupled to the server platform, a "DB2" relational database management system (RDBMS) executing on the server platform, said method steps comprising:

(1) at the server platform, receiving an input `args` parameter being of a "sqlda" data type, said input `args` parameter comprising a "sqlvar" parameter defined by said sqlda data type, said sqlvar parameter comprising a pointer pointing to a string that was transferred from the client platform to the server platform, said string

being a string representation of a file located on the client platform;

(2) at the server platform, using said pointer in said sqlvar parameter of the input_args parameter to access and retrieve said string;

(3) at the server platform, writing said string to a new file.

28. The program storage device of claim 27, wherein said input_args parameter comprises a second sqlvar parameter having a pointer that points to a file name of said file, and a third sqlvar parameter having a pointer that points to a target path string, said method steps further comprising the steps of:

(4) at the server platform, using said pointer in said second sqlvar parameter of the input_args parameter to access and retrieve said file name; and

(5) at the server platform, using said pointer in said third sqlvar parameter of the input_args parameter to access and retrieve said target path string;

wherein step (3) comprises the step of:

at the server platform, writing said string to a new file using said file name at a location in said server platform identified by said target path string.

29. A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform method steps for enabling an user to distribute a procedure from a client platform to a plurality of server platforms, and for enabling the user to make and register the procedure at the server platforms, a "DB2" relational database management system (RDBMS) executing on each server platform, said method steps comprising:

(1) at the client platform, displaying a list of servers;

(2) at the client platform, enabling the user to select any of said servers;

(3) at the client platform, commanding a "client application enablement" (CAE) module to distribute a file associated with the procedure to the selected servers, the CAE module being resident in the client platform and representing a client component of the DB2 RDBMS; and

(4) at the client platform, commanding the CAE module to cause the selected servers to make and register the procedure;

wherein step (3) comprises the steps of:

(a) converting the file to a string;

(b) setting a procedure_to_invoke parameter equal to information identifying a file transfer procedure located at each selected server;

(c) causing a pointer in a first "sqlvar" parameter to point to said string, and causing a pointer in a second sqlvar parameter to point to a file name of the file, the first and second sqlvar parameters being part of an input_args parameter, the input_args parameter having a "sqlda" data type, the sqlda data type being suitable for passing scalar values to procedures; and

(d) invoking for each selected server a "sqleproc" function provided by the CAE module, a parameter list of said sqleproc function comprising said input_args parameter and said procedure_to_invoke parameter, said sqleproc function when executed causing said input_args parameter to be passed to said file transfer procedure at each selected server, and also causing said file transfer procedure to be invoked at each selected server.

30. The program storage device of claim 29, said method steps further comprising the steps of:

- (5) at each server, receiving the input_args parameter;
(6) at each server, using said pointer in said first sqlvar parameter of the input_args parameter to access and retrieve said string;
(7) at each server, using said pointer in said second sqlvar parameter of the input_args parameter to access and retrieve said file name; and
(8) at each server, writing said string to a new file in said each server using said file name.
31. The program storage device of claim 29, wherein step (4) comprises the step of:
- setting the procedure_to_invoke parameter equal to information identifying a make function located at each selected server; and
- invoking for each selected server the "sqleproc" function, said sqleproc function when executed causing each selected server to invoke the make function, the make function when executed operating to make the procedure.
32. The program storage device of claim 29, wherein step (4) comprises the step of:
- setting the procedure_to_invoke parameter equal to information identifying a registration function located at each selected server; and

invoking for each selected server the "sqlproc" function,
said sqlproc function when executed causing each
selected server to invoke the registration function, the
registration function when executed operating to reg-
5 ister the procedure with the DB2 RDBMS at said each
selected server.

33. The method of claim 1, wherein said file is a vector
file.

10 34. The method of claim 2, wherein said file is a vector
file.

35. The method of claim 6, wherein said file is a vector
file.

15 36. The system of claim 12, wherein said file is a vector
file.

37. The program storage device of claim 22, wherein said
file is a vector file.

20 38. The program storage device of claim 23, wherein said
file is a vector file.

39. The program storage device of claim 27, wherein said
file is a vector file.

* * * * *

~~ADD A2~~ →

ADD B1 →

652649 223363

5

10

15

30

* * * * *